MECH60017/MECH96014/MECH96038 STATISTICS
MATLAB TUTORIAL SHEET III

The following exercises are to develop your skills in MATLAB and the Statistics Toolbox for statistical data analysis. You should attempt all these exercises in order to ensure that you are familiar with the statistical functions in MATLAB for the coursework that will be issued in Spring Term.

1 Building your own random number generator

It is possible to build your own versions of `binornd`, `normrnd`, `exprnd`, etc using just the basic `rand` and `randi` commands in MATLAB. As you saw in MATLAB Exercises II, the `rand` command generates a random number between 0 and 1, and the `randi(imax)` command generates a random integer between 1 and `imax`.

Suppose we want to draw random numbers according to a random distribution with cumulative distribution function $F(x)$ but only have access to a uniform random number generator. How do we proceed?

1.1 Inverse Transform Sampling

If the inverse $F^{-1}(x)$ of the c.d.f. is available then the **inverse transform sampling** approach can be used:

1. Generate a random number between 0 and 1, u - ie, u is from a $\text{Uniform}(0,1)$ distribution.
2. Calculate $x = F^{-1}(u)$ - x is taken to be a random sample from the distribution with c.d.f. $F(x)$.

This approach is usually fine for discrete probability distributions where it is easy to compute the c.d.f. $F(x)$ and continuous distributions where the integrals required for the c.d.f. $F(x)$ are availiable.

<u>Tasks</u>

Use the inverse transform sampling method to build your own functions to generate random samples from some standard distributions.

1. Exponential distribution, $X \sim \text{Exp}(\lambda)$.

   You choose the value of the parameter, and you decide the number n of random samples to generate - try the following for a few different values for n.

(a) Generate u, a random number between 0 and 1, and calculate $x = F^{-1}(u)$. (Note you should explicitly derive F^{-1} for the exponential distribution.) Repeat for n random samples from an $\text{Exp}(\lambda)$ distribution.

(b) Use a Q-Q plot (`qqplot(X, PD)`) to check your method by plotting the quantiles of your random numbers against the theoretical exponential quantiles. Use `makedist` for the theoretical distribution.

(c) How does sample size affect the Q-Q plots?

2. Poisson distribution, $X \sim \text{Poisson}(\lambda)$.

   MATLAB actually has inverse functions, eg `poissinv`. Repeat the tasks above for the Poisson distribution, using `poissinv` instead of deriving an expression for F^{-1}.

   Is the Q-Q plot appropriate to check both the Poisson and exponential distributional assuptions?

1.2 Box-Muller Method

The inverse transform sampling method cannot be used to generate normally distributed random numbers as the c.d.f. for the normal distribution is not available analytically.

The **Box-Muller** method provides a method for generating normally distributed random variables:

1. Generate two uniformly distributed random numbers between 0 and 1, u_1 and u_2.
2. Calculate $z = \sqrt{-2 \ln u_1} \cos(2\pi u_2)$.

z is then a sample from a normal distribution with mean 0 and variance 1. The transformation $x = \sigma z + \mu$ will give an observation from a $\text{N}(\mu, \sigma^2)$.

<u>Tasks</u>

1. Use the Box-Muller method to build your own function to generate random samples from the normal distribution with mean μ and variance σ^2 (you choose numerical values for μ and σ^2).
2. Use a Q-Q plot (`qqplot(X)`) to check that your method by plotting the quantiles of your random samples against the theoretical normal quantiles.